Automated compliance checking using building information models

David Greenwood[1], Stephen Lockley[1] Sagar Malsane[1], and Jane Matthews[1]

[1]School of the Built Environment,
Northumbria University, Newcastle-upon-Tyne, NE1 8ST,
United Kingdom

Email: *david.greenwood@unn.ac.uk; stephen.lockley@unn.ac.uk; sagar.malsane@unn.ac.uk; jane.matthews@unn.ac.uk.*

Abstract:
Building designs in the UK are currently checked manually against a frequently changing and increasingly complex set of building regulations. This is a major task for both designers and enforcers, often leading to ambiguity, inconsistency in assessments and delays in the overall construction process. Technical developments in Building Information Modelling (BIM) offer the potential for a new generation of software tools that can automate the checking of compliance with building codes, thus improving the efficiency of building design and procurement. To attain these efficiencies designers must change their working practices and move away from the definition of a building in multiple and disparate documents to a single coherent building model from which the documentation is generated. Theoretically, this building model could contain sufficient information to respond to interrogation at the level of building code compliance, though in practice only a percentage of the required information is normally present. This paper reviews previous research into automated code compliance, identifies the key issues for future development and examines the causes of information paucity for compliance checking in the current generation of BIM tools.

Keywords:
Automated compliance checking, BIM, UK Building Regulations.

1 Introduction

With the development of Building Information Modelling (BIM) software automated compliance checking of building designs using model checking software (MCS) is becoming a realistic prospect (see Choi and Kim, 2008). It is likely that in the near future BIMs will become important digital assets that are not only key instruments in communicating design but also in obtaining approval from statutory bodies (see, for example, Raslan and Davies, 2010). The manual checking of building designs for compliance against national codes is complex and prone to human error with significant cost implications. It is claimed that automated compliance checking would not only prove beneficial to designers but to also building certifiers, consultants, building code authorities, specification writers and builders (Tan *et al.,* 2010).

However, automated compliance checking requires the application of software tools, which are normally generic and international, to codes and regulations, which are specific and local. To date, the strategy adopted in most compliance checking initiatives has been to convert proprietary BIM models into the international standard format (IFC) and then to author bespoke compliance rules that can be executed using this model. The problem with this approach is that the international BIM tools do not populate these IFC models with the all the required data; in particular, the relationships with local codes and regulations. Thus, for reliable compliance checking, additional data must be provided by the design team as a separate activity.

As part of an ongoing funded research project the authors propose a development of this strategy by defining, within the IFC model, a domain extension for England and Wales Building Regulations using the established methodologies of the BuildingSMART Alliance (http://www.buildingsmart.com/). The objectives are to work with the National Building Specification organisation to identify concepts, objects and properties that are entrained in the Building Regulations (England and Wales) and explore a range of formal syntaxes for creation of the requisite rules.

2 Technological development

A situation that is analogous to the uptake of BIM is that of the development and early adoption of computer aided design (CAD) in the last decades of the previous century.

The adoption of CAD

During the 1970's and 1980's two-dimensional (2D) computer aided design was developed and deployed by the 'early adopters' amongst construction design practices. By the end of the 1990's 2D CAD was used in the majority of construction design activities and 3D design systems were available, but their uptake was limited. One of the main reasons for the resistance to the use of 3D CAD in construction was the lack of perceived benefits. Essentially, this generation of software tool was drawing-oriented, i.e. the underlying representation in the tools was graphical (in terms of lines, arcs, points, *etc*.): a door, represented in this way would therefore 'behave' like a series of graphical objects, and not a door. For example, changing the opening width of the door meant making a line or arc shorter or longer. Whilst this was acceptable in 2D CAD (being no different to traditional paper-based drawing procedure) in 3D CAD the amount of graphical change required was significantly greater. Thus, although 3D CAD brought great potential benefits (such clash detection and visualisation) the overhead of authoring the models rendered these benefits cost-ineffective. The end result was that most design practices used 3D for *presentational* purposes and disposed of the model once this was complete in favour of traditional 2D drawings.

The initial response of the CAD vendors to resolve this was *parametric object design*: rather than the user having to define the lines and arcs of the door they would be automatically generated from a set of parameters such as height and width. As these parameters were altered so were the resulting graphical representations. This approach effectively accelerated the process of authoring the graphical representation of the building, but essentially the resulting model was still a graphics-oriented model designed to be output as drawings.

A paradigm shift

Concurrently, during the 1990s computer scientists were making a paradigm shift in the way software was designed and authored. This involved moving away from a functional or procedural way of conceiving software to an object-oriented paradigm. Object-oriented (OO) programming required software engineers to construct their systems in terms of the real world objects that were involved in the problem to be solved. This shift in focus was intended to lead to more stable and maintainable software solutions that could be understood by domain experts who were intended to use the systems.

Accordingly, construction industry researchers began to adopt the OO approach to software design and these resulted in several research prototypes (see, for example, Stumpf *et al.,* 1996). By the mid 1990s, the major CAD vendors were adopting this approach however, it became apparent that building designs comprise thousands of different types of real world objects and that it was a task beyond any individual company to create computer models for all of these. There was a need for international standardisation for objects in building models and this led to the beginning of the development of the Industry Foundation Classes (IFCs). It is important to note that this initiative to develop object and product models was not unique to the Construction Industry, but part of an international initiative - the Standards for Product Data Exchange (STEP) - for all sectors of industry (see Pratt, 2001). This pan-industry initiative initially focussed on developing a modelling language (EXPRESS) and file exchange format (ISO 10303 part 21) and these are the cornerstones of the current IFC implementation. These steps enabled the next phase of development, namely, generic resources or libraries that could be shared across all sectors to avoid duplication and accelerate development.

3 Strategies and approaches to system development

Approaches to developing automated building code-checking have been reported in the literature for the last two decades (Eastman *et al*., 2009).

3.1 Singapore (CORENET)

The BP-Expert system had been available in Singapore from as early as 1995 for checking 2D drawings with a view “to reengineer and streamline the fragmented work processes in the construction industry, so as to achieve quantum improvements in turnaround time, quality and productivity” (Evelyn and Fatt, 2004, p.1). In 2000 it was replaced by e-PlanCheck as part of the Construction and Real Estate NETwork (CORENET) project (Sing and Zhong, 2001). CORENET was one of the first initiatives in automated code-checking, and was funded by the Singapore Ministry of National Development and carried out by the Construction and Real Estate Network (Choi & Kim, 2008). This aimed to provide an internet based electronic submission system for checking and approving building plans. Building proposals were submitted as a combination of existing 2D drawings with additional information provided in supplementary IFC-based files. The system utilised many of the convergent technologies described in the previous section (OO software design, STEP and BIM); it was considered to be ‘cutting edge’ and conceptually strong, yet there is little evidence of continuing work on the specific initiative.

The aim, as before, was to improve performance, increase coverage and check compliance of building data in an IFC format. However, while the implementation of the IFC by CAD vendors remained focused on geometry many of the requirements for compliance checking were not available. E-PlanCheck addressed this by commissioning an independent platform, FORNAX, to sit on top of the already existing EDM ModelChecker. FORNAX is an object library written in C++. Each object contains all the relevant attributes for the Singapore codes as well as the rules that apply to that object. Each object is designed to be extensible in order to cover the requirements of other countries, and as a result CORENET e-PlanCheck was used as the basis for pilot projects in Norway, New York and Australia (Khemlani, 2005). Despite ongoing attempts to implement performance based checking, reported difficulties with verifying data quality (Solihin, 2004), and its inability to support the checking of design standards throughout the different design stages of the project (Ding *et al.,* 2006) e-PlanCheck in Singapore is still the only system that is currently operational.

3.2 Norway (Statsbygg)

The CORENET work was developed and emulated in Norway with the ByggSok system (Haraldsen *et al*., 2004). This is an e-Government system comprising three modules: an information system, a system for e-submission of building applications and a system for zoning proposals. Driven by the Norwegian Building and Construction industry and supported by Standards Norway and Norwegian buildingSMART it is heavily based on IFC standards. The work is ongoing and currently focussing on the issues of classification, terminology and standardising rule-checking in construction at an international level.

Building upon their e-PlanCheck pilot projects Norwegian developers (Statsbygg) have experimented with multiple systems as part of their efforts to extend the use of IFC to the entire project life cycle in support of their mandate that by 2010 all properties will use IFC based BIM (Sjøgren, 2007). The resulting systems have been piloted on real projects, with data being exchanged through a wide selection of software to suit the various stages / tasks of the project lifecycle. On the HITOS pilot, the code checking efforts have focused predominately on accessible design. Here the building model data are stored and accessed through EDM Model Server in IFC format. The accessibility rules are parameterised, mapped to their associated building objects and executed using Solibri Model Checker's Constraint Set Manager. Solibri communicates directly with building model data in IFC format, but retrieves only the objects it needs – i.e. those mapped to the accessibility rules. The rules implemented to date focus predominantly on geometrical constraints and as such the objects and parameters are supported by the IFC data models produced by current BIM packages. The Statsbygg Solibri system does not support the enhancing of these data models or the export to IFC format, and so cannot currently be used for compliance checking of attributes not supported by the current BIM vendors. The Solibri Constraint Set Manager is implemented in java and ships with a library of built-in parameterised rules which can be configured by adjusting the parameters. New rules, however, must be custom made in collaboration with the Solibri software developers and as such are not easily adapted for other software. Solibri has the benefit of powerful 3D modelling engine which, in combination with the ability to directly read IFC files, allows for clear visual reporting of rule infringements for the user. Solibri's built-in rule library contains rules for validating a data model prior to rule checking which is useful.

3.3 Australia (DesignCheck)

Both the Solibri Model Checker and Express Data Manager were considered as possible platforms for automated code checking in Australia (Ding *et al*., 2004) again focusing on accessible design regulations. The work was undertaken by Commonwealth Scientific and Industrial Research Organisation (CSIRO) and the University of Sydney and was funded by Australia's Cooperative Research Centre for Construction Innovation. EDM was eventually selected and the resulting automated code checking system – DesignCheck is currently on trial by the construction industry in Australia (Eastman *et al.,* 2009).

Design Check uses object based rules, encoded using EDM. Building data models, in IFC format, are imported into the EDM database and transformed into the Design Check internal model. The Design Check model includes building code specific information not currently implemented by BIM vendors. A mapping schema, written in ExpressX translates the building data model from IFC format into the DesignCheck schema. The strategy is similar to that of e-PlanCheck in Singapore; however, DesignCheck has the advantage of supporting the ability to check for compliance at various stages in the design process, as it has a rule schema for early and detailed design stages as well as for specification. It is therefore targeted at Architects and Designers rather than just Building Control certifiers (Ding *et al*., 2006). As yet DesignCheck does not have the ability to view 3D models and all reports are text based.

3.4 The United States (General Services Administration and International Code Council)

Similar work on code-checking began in the United States around 2000, with the initial emphasis on health, safety and welfare. A major driver of BIM and validation of BIM models in the United States is the US General Services Administration (GSA). The GSA issued BIM-guidelines in late 2006 (GSA, 2006) and in 2007 proposed that all planners seeking funding for their spatial planning projects would need to produce BIM models for validation as an open standard (GSA, 2007).

The aptly named *SmartCodes* is a project driven by the International Code Council, in conjunction with AEC3 and Digital Alchemy. This project has focused largely on addressing the problem of transforming paper-based codes (of which there are thousands) into machine-interpretable rules; generally a lengthy process requiring many iterations between Building Code officials and software developers. In order to streamline this process the SmartCodes project developed a methodology for applying tags to electronic copies of Building Codes using a 'tag dictionary', or ontology (Wix *et al*., 2008). The rules are then automatically extracted, following a strict mathematical pattern, into an IFC constraints schema. The resulting IFC constraints schema is mapped to the IFC building data model via the tag dictionary. The rules can currently be executed using either Solibri Model Checker, or AEC3 XABIO. The SmartCodes project does not support building code specific information that is not currently implemented by BIM vendors (Eastman *et al.* 2009).

4 Conclusions and implications for future work

The research to date has identified four key requirements for automated code checking

1) Computer programmed rules must be easily understood by Regulation authors;
2) The lifecycle of the rule base must be independent of software and schema updates;
3) All development must be compliant with Open Standards;
4) Consideration must be given to the industry processes of model authoring.

These will now be considered in turn, and at each stage the implications for the development of an effective automated checking system for UK building regulations will be considered, alongside what is currently available in the world of BIM systems.

4.1 Easily-understood and accessible rules

Several of the initiatives outlined above have focussed on creating rules and mapping the entities encapsulated in these rules to the international building model schema. This schema is designed to support the needs of an international user and takes little consideration of national semantics (e.g. UK practice and culture). The result is that the authors of the regulations will be required to accept mappings that lie somewhere between the concepts in UK practice and their abstract counterparts in the IFC schema. For example a water closet (WC) is a well-understood description in the UK but in the IFC schema may well be represented as a 'sanitary flow inlet'. Rules derived from the regulations that are mapped to this IFC schema will be difficult for the authors to understand and in the long term will make maintaining the rule base complex.

Consequently, and following the FORNAX approach (see above) we intend to develop a UK-specific building model schema, embodying concepts that fit with UK custom and practice; mappings will be created between the UK schema and the IFC schema through the domain extension approach, ensuring interoperability and maintainability. It is intended that these mapping will have long term durability. Authors will define rules in terms of the UK schema ensuring comprehensibility and maintainability. Since work on the FORNAX system began there have been significant developments in the IFC schema as well as computing languages that can implement it. These developments make it timely to revisit the FORNAX approach.

4.2 Independence of software and model schema updates

The core information models in the current generation of BIM software tools are designed to support international markets and are intentionally not localised to national jurisdictions by the vendors. It is simply not economically viable for the major CAD vendors to develop multiple local flavours of their product. Rather, their strategy is for localisation to be applied by third parties to comply with specific legal and cultural requirements. Clearly, though, it is not efficient for a design practice operating in both Scotland and England for example, to have to use different BIM tools for local projects. In contrast, building regulations and codes are highly specific to a location and a culture often only applicable to small regions. The UK, for example, has three different regional implementations (shortly to become four with the separation of Welsh regulations from the English). In addition, the development cycles for regulations and software are driven by fundamentally different demands: in the case of software, by market needs and technology changes; and in the case of regulations, by changes in the law, consultation processes, and extraordinary events. It is essential that any automated regulation-checking solution can be updated independently of the development cycle of the software tools used.

4.3 Compliance with Open Standards

This being the case, the implication that the rule-checking tool must be independent of the software tool points naturally to the use of an open standard for communication. There are several Open Standards for Building models but our research to-date has suggested that the most comprehensive, for the purpose of regulatory control, are the IFCs. Other standards such as CityGML and GBXML are targeted at a specific use and do not model the breadth of concepts required. The IFCs are also widely adopted by the major CAD vendors and are generally accepted as the standard most likely to succeed.

4.4 Consideration of the industry processes of model authoring

So far we have considered the issues of the rule checking schema, but consideration is also needed as to how building models are authored by the design team. There is currently no *single* software application that can fully populate a building model. Indeed it is arguable whether a single comprehensive and coherent building model is a realistic aspiration. At present, the working processes and issues of ownership and liability in the authoring of building design works against the creation of such a single fully-populated model (Greenwood *et al.,* 2010). The approach currently taken is to define views for data exchange between software tools and organisations. Currently the most widely used and agreed view is the ‘Project Coordination View’; others are under development. Essentially, the purpose of the view is to set out formally what information must be present in the building model for it to be considered fit for purpose. The difficulty is that views can only contain entities and properties that are currently defined in the IFC schema. Clearly there are some entities and properties that are specific to UK custom and practice that would never be defined for international usage. Whilst there is a mechanism to work around this problem in terms of data exchange (in the form of IfcPropertySets) there are unlikely to be any widely used software tools that will help all members of the design team author these data.

The overall implication of this is that one of the widely used international CAD tools will be used to author the majority of the building model (elements, materials, geometry etc) and extension programmes will be needed to author data that are not provided.

Part of the current research programme will be directed at investigating optimal approaches to creating these extension programmes. Two strategies are under consideration for identifying data that would be required for automated code-checking but would not normally be present in the model. *Constraint-based* analysis executes the compliance rules on the building model and reports when insufficient information is available to determine compliance. *Schema-based analysis* compares the structure of the building model data with the structure of the UK Building Regulations Schema to determine the required additional data inputs. It is hoped that from this analysis a data entry tool can be generated that is specific to the given building model. Thus the approach would be extensible to any building model, model view or localised schema and allows flexibility in the building model authoring process.

5 Acknowledgements

The authors gratefully acknowledge the co-funding provided for this research by the Royal Institute of British Architects Enterprises Ltd.

6 References

Choi, J., and Kim, I. (2008) ‘An Approach to Share Architectural Drawing Information and Document Information for Automated Code Checking System’, *Tsinghua Science & Technology*, **13** (1), pp 171-178.

Ding, L., Drogemuller, R., Jupp, J., Rosenman, M. and Gero, J. (2004) ‘Automated Code Checking’ *CRC CI International Conference*, Gold Coast, Australia. Available at: http://en.scientificcommons.org/50758591 (viewed: 04 June 2010)

Ding, L., Drogemuller, R., Rosenman, M., Marchant, D and Gero, J. (2006) ‘Automating code checking for building designs’. *The Second International Conference - Clients Driving Innovation: Moving Ideas into Practice*, The Cooperative Research Centre (CRC) for Construction Innovations, Gold Coast, Australia, March 2006. Available at: http://web.arch.usyd.edu.au/~john/publications/2006/06Ding_etalCRC06.pdf (viewed: 04 June 2010)

Eastman, C, Lee, J-m, Jeong, Y-s and Lee, J-k (2009) ‘Automatic rule-based checking of building designs’ *Automation in Construction*, **18** (8) Pages 1011-1033.

Evelyn, T. & Fatt, C. (2004) ‘Building Smart - A strategy for implementing BIM solution in Singapore’. Available at: http://www.itsc.org.sg/pdf/5_BIM.pdf (viewed: 02 June 2010).

Greenwood, D., Lewis, S. Lockley, S. (2010) ‘Contractual Issues in the Total Use of Building Information Modelling’, in Barrett, P., Amaratunga, D., Haigh, R., Keraminiyage, K. & Pathirage, C. (eds) *Proceedings of CIB 2010 World Congress ‘Building a Better World’*. Salford, May 2010. The University of Salford.

Haraldsen, M, Stray, TD, Päivärinta, T, and Sein, MK (2004) ‘Developing e-Government Portals: From Life-Events through Genres to Requirements’ in Rolland, KHR (ed.), *Proceedings of 11th Norwegian Conference on Information Systems,* Stavanger, pp.44-70.

Khemlani, L (2005). ‘CORENET e-PlanCheck: Singapore's Automated Code Checking System’. AECbytes, October, 2005 [Online]. Available at: http://www.aecbytes.com/buildingthefuture/2005/CORENETePlanCheck.html (viewed: 02 June 2010).

Pratt, MJ (2001) ‘Introduction to ISO 10303—the STEP Standard for Product Data Exchange’, Journal *of Computing and Information Science in Engineering* 1 (1) 102-4.

Raslan, R. and Davies, M (2010) ‘An analysis of industry capability for the implementation of a software-based compliance approach for the UK Building Regulations 2006’, *Building Services Engineering Research and Technology*, **31**:141

Sing, T. F. & Zhong, Q. (2001) ‘Construction and Real Estate NETwork (CORENET)’, *Facilities*, **19**, pp. 419-428.

Sjøgren, J. (2007). ‘BuildingSMART –a smart way for implementation of standards’ [Online]. Available at: http://www.plcsresources.org/papers/s1000d/day3/buildingSMART%20-%20Jons%20Sjogren.pdf (viewed: 29 June 2010)

Solihin, W. (2004) 'Lessons learned from experience of code-checking implementation in Singapore - Success, Challenges, and Future Outlook' [Online]. Available at: http://www.buildingsmart.com/files/u1/ned_20from_20experience_20of_20code_checking.pdf (viewed: 02 June 2010).

Stumpf, R., Ganeshan, R., Chin, S., and Liu, L.Y. (1996) 'Object-oriented model for integrating construction product and process information', *Journal of Computing in Civil Engineering*, **10** (3), pp. pp. 204-212.

Tan, X., Hammad, A., and Fazio, P. (2010) 'Automated Code Compliance Checking for Building Envelope Design', *Journal of Computing in Civil Engineering*, **24** (2), pp. 203-211.

United States General Services Administration (2006) '02 – GSA BIM Guide for Spatial Program Validation Appendices, GSA BIM Guide Series 02' [Online]. Available at: http://www.gsa.gov/graphics/pbs/GSA_BIM_02_Appendix_v09.pdf (viewed: 08 June 2010).

United States General Services Administration (2007) 'BIM Guide for Spatial Program Validation, GSA BIM Guide Series 02' [Online]. Available at: http://www.gsa.gov/graphics/pbs/BIM_Guide_Series_02_v096.pdf (viewed: 08 June 2010).

Wix, J. Nisbet, N. and Liebich, T. (2008). 'Using constraints to validate and check building information models'. *EWork and EBuisness in Architecture, Engineering and Construction: ECPPM 2008*. London, pp. 467- 475.